Managing Distributed Data

Heather Paskevic

Agenda

- **Data Distribution**
- **Versioning**
  - **Version management**
  - **Conflict detection**
- **Geodatabase Replication**
  - **Workflows**
  - **Replica creation**
  - **Synchronization**
  - **Schema changes**

Data Distribution

- **Copies of data distributed in multiple locations**
  - Between two or more geodatabases
- **Can provide**
  - Improved data availability with poor networks
  - Load balancing: Separate offices can work on same data
  - Field projects
  - Fail over
  - Contractor/Moble use

Data Distribution– Use Cases

- **Mobile users and field crews who need to be disconnected from the network**

- **Users who need to maintain copies of data at different organizational levels (city, county, state)**

- **Users who want to maintain copies of data at different geographic facilities**

- **Users who need to distribute work to contractors**

- **Production and publication geodatabases**

# Versioning Overview

Understanding Versioning

- **What is a version?**
  - Represents a snapshot in time
  - Isolates user's work across multiple edit sessions, no locking
  - Always used to access ArcSDE geodatabases
  - Includes methods for merging changes from multiple versions and conflict detection
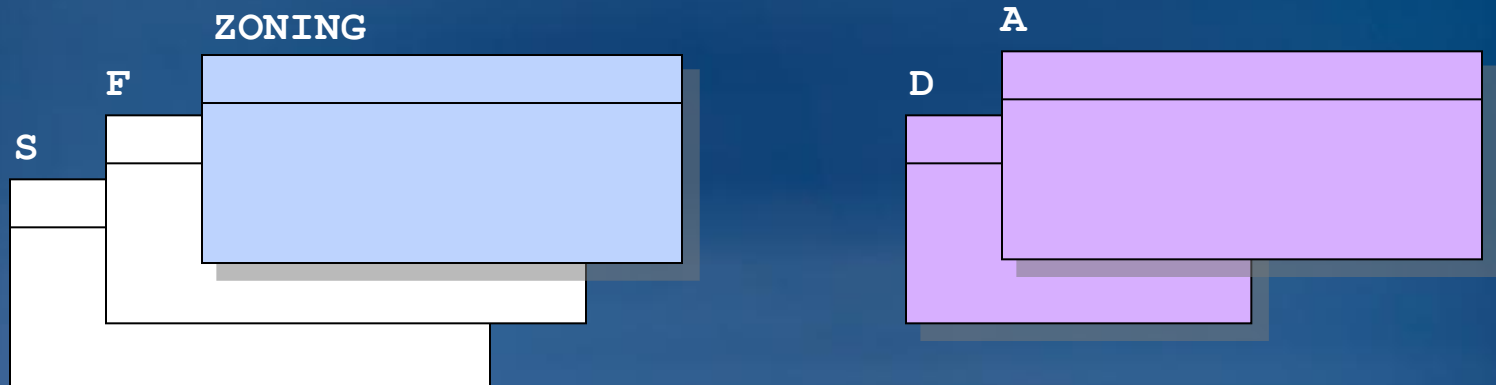 - Entire geodatabase is visible through every version

USER1 version

Original version

USER2 version
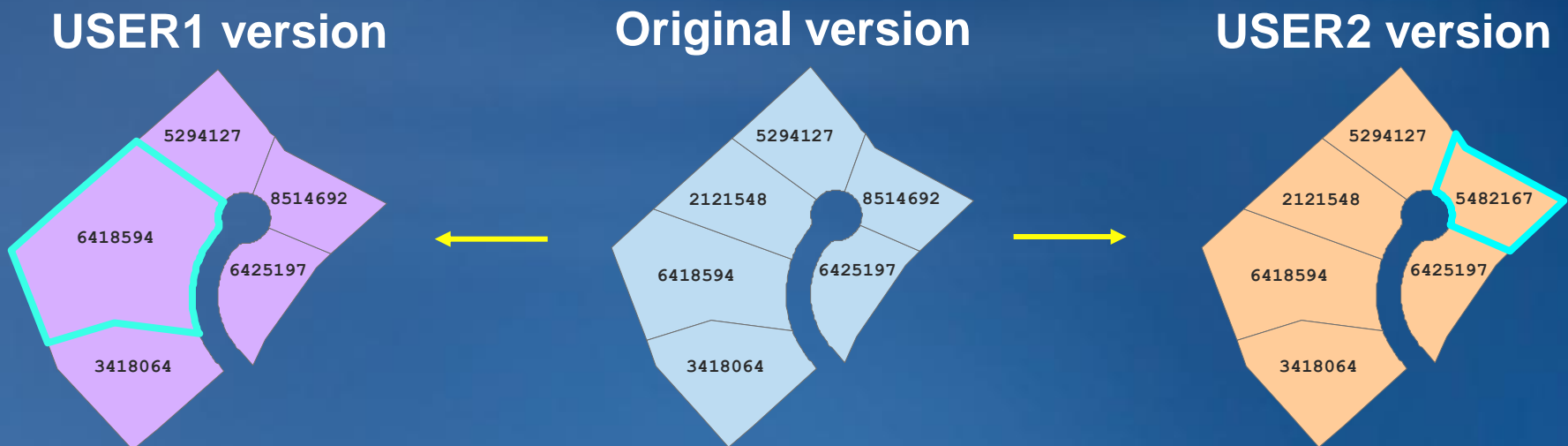
Registering as Versioned

- **Enables versioned edits**
  - **Feature classes, feature datasets, tables**
- **Creates delta tables**
  - **Adds (A) and Deletes (D)**
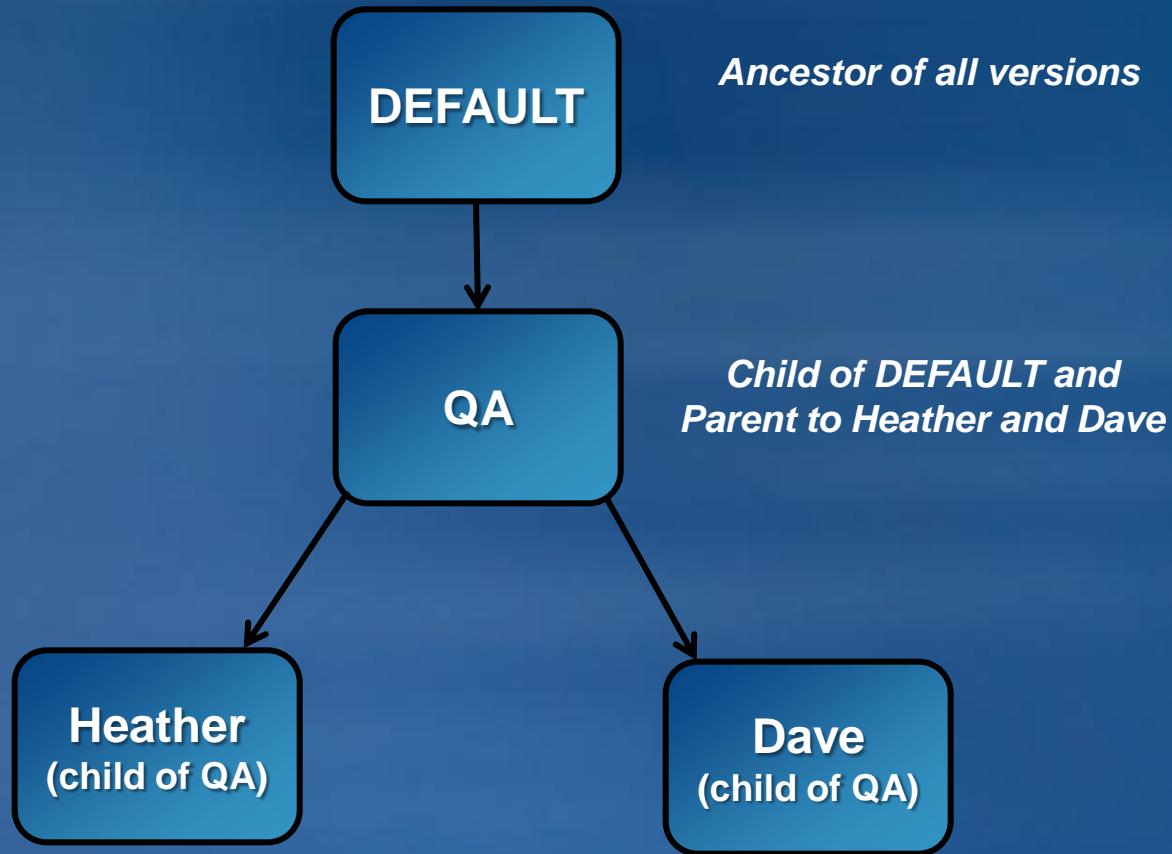 - **Registering does not create a version**
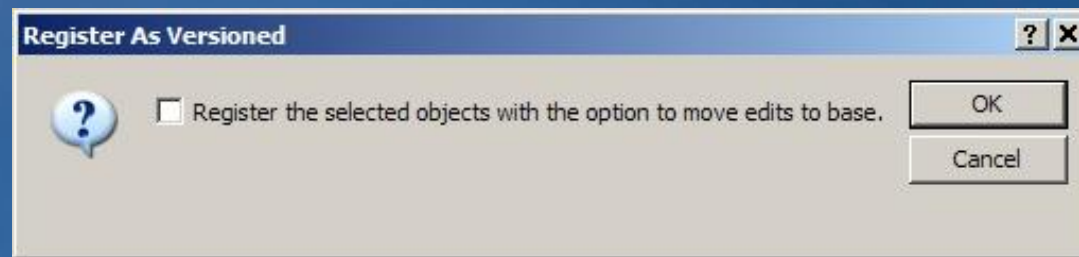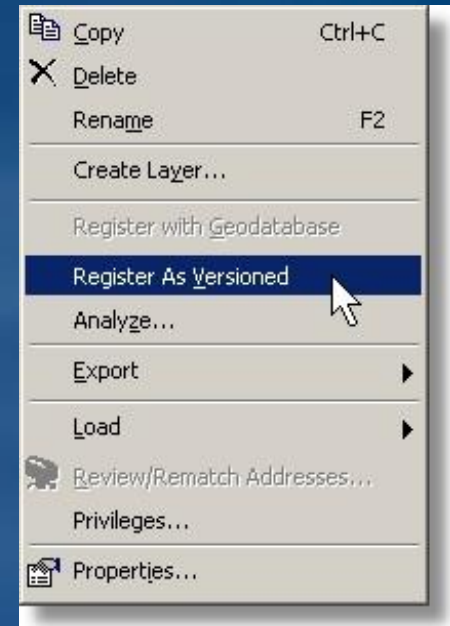
Registering as Versioned: Delta tables

- **Delta tables are created when object is registered as versioned**
- **Changes are stored in delta tables**
  - Adds table records inserted and updated rows
  - Deletes table records deleted and updated rows
  - Preserves changes made by all users
- **Owner of delta tables is also owner of multiversioned table**

Version Creation

DEFAULT — *Ancestor of all versions*

QA — *Child of DEFAULT and Parent to Heather and Dave*

Heather (child of QA)

Dave (child of QA)

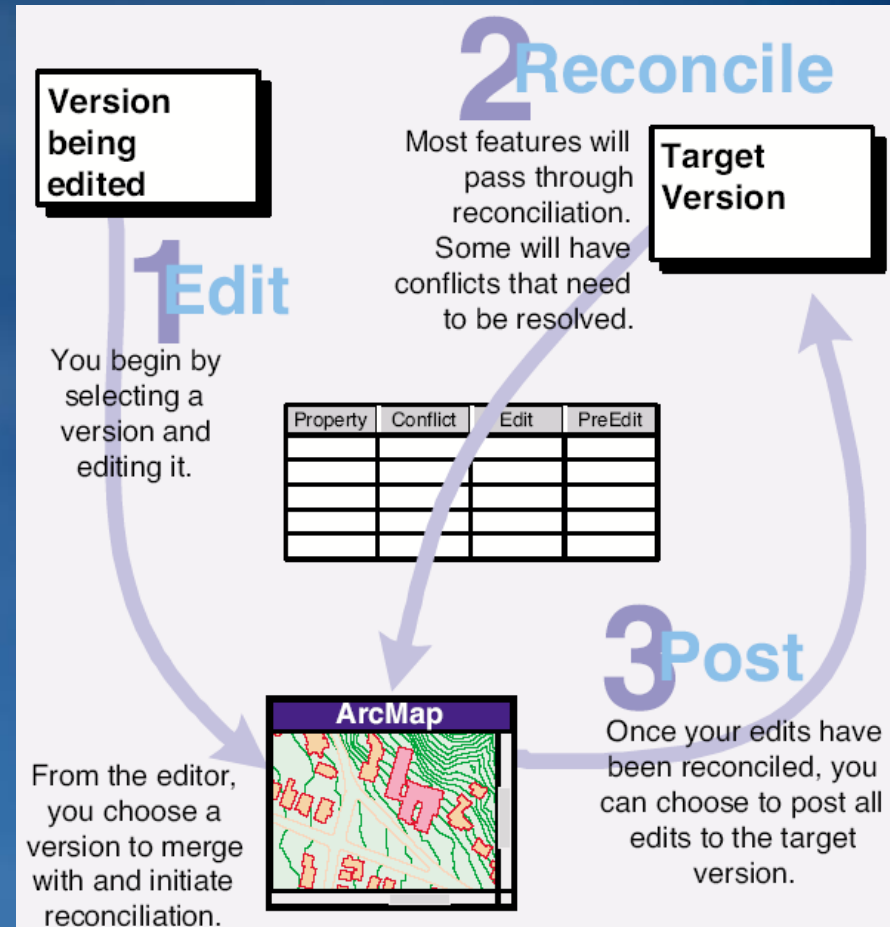*Versioning does not create a copy of the data!*

Demonstration

- Create a new version
- Add the Versioning toolbar, explore the versions
- Edit data in a version

Merging versions

- **Child version is reconciled with parent**
  - Target version is any version in direct ancestry
 - Detects differences and discovers conflicts

Conflict Detection

- Edit conflicts are typically not frequent
- Conflicts arise when same feature is modified
  - Update / update or update / delete
- Detected on save and reconcile
  - Save not completed until conflicts are resolved
- Choose to abort or resolve conflicts
  - By default, other users' edits prevail

Resolving Conflicts Between Versions

- **Conflicts resolved either:**
  - Manually
  - Based on predefined settings
- **Detected on reconcile**
- **When conflicts found, warning is returned**
  - Yes – Opens Conflicts window
  - No – Conflict resolution defined on Reconcile dialog box
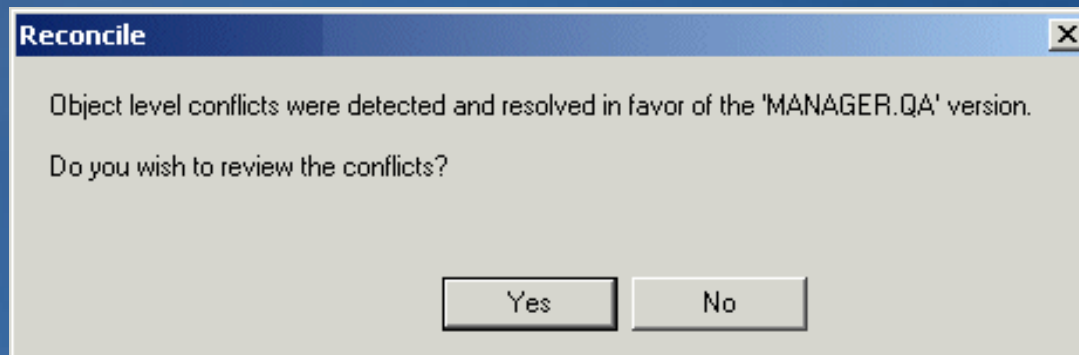
Viewing Conflict Choices
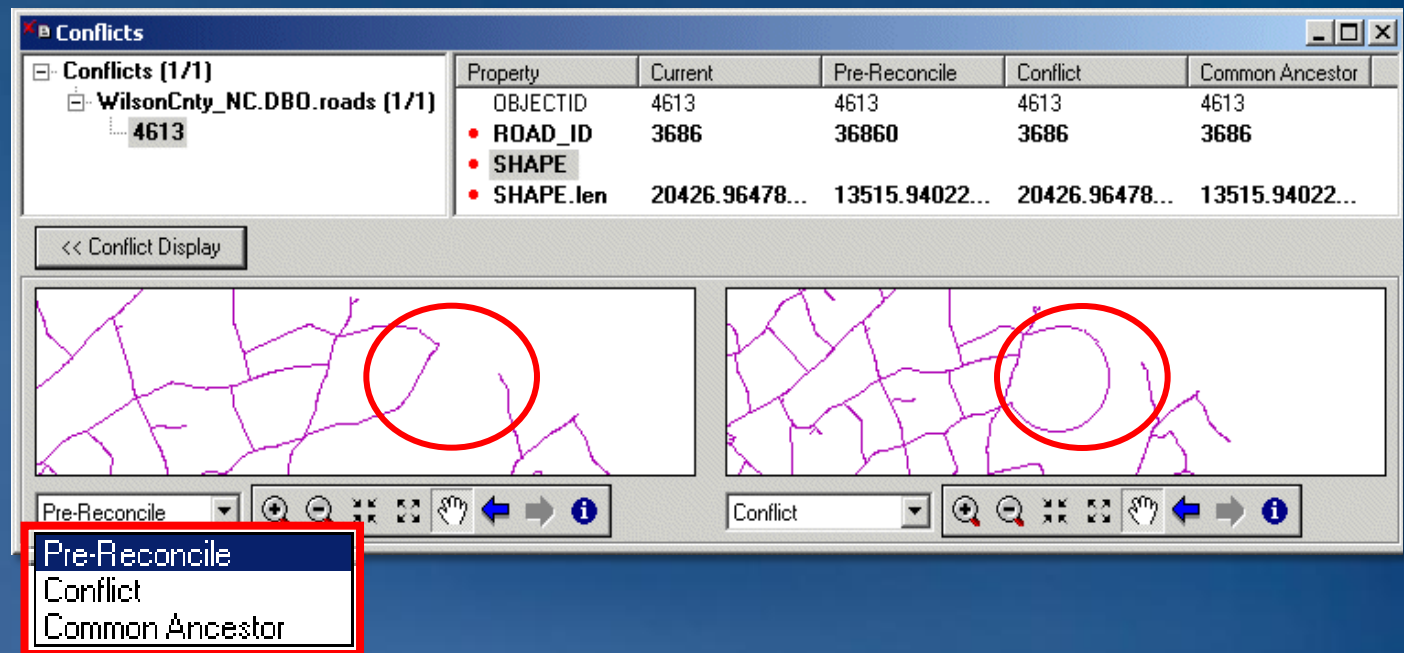
- **Pre-reconcile – Your edit**
- **Conflict – Edit made by another user**
- **Common ancestor – Representation before either edit**

Post

- **Synchronizes current edit session with target version**
- **Must be done in an edit session**
- **Changes saved to target**
  - **Two versions now identical**
  - **Cannot be undone**
- **Performed after reconciling with another version**
- **Not necessary to post after reconcile**
 - **Depends on workflow**

Demonstration

- **Reconcile and Post**
- **Conflict Detection and Resolution**
- **Geodatabase Toolset (GDBT)**
  - **Tools to assist you with monitoring, investigating and reporting the performance of a multiuser geodatabase**
 - http://www.esri.com/software/arcgis/extensions/gdbt/index.html

Introducing Geodatabase Replication

- **Copies of data across geodatabases**
  - Supports the full geodatabase data model
- **Works across different DBMS types**
- **Users can work on the same data in separate locations**
- **Changes can be synchronized between geodatabases**
- **Built on top of the _versioning_ architecture**
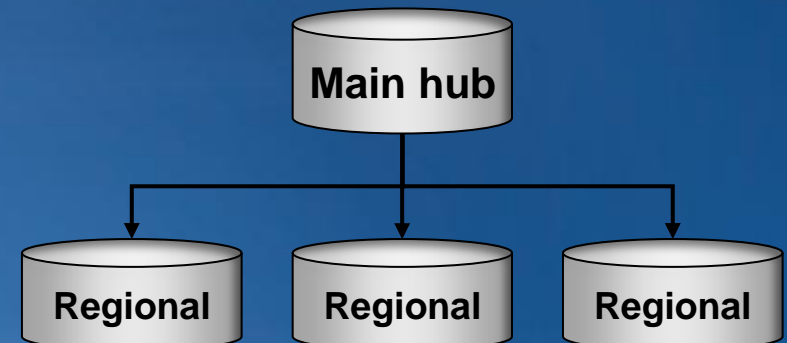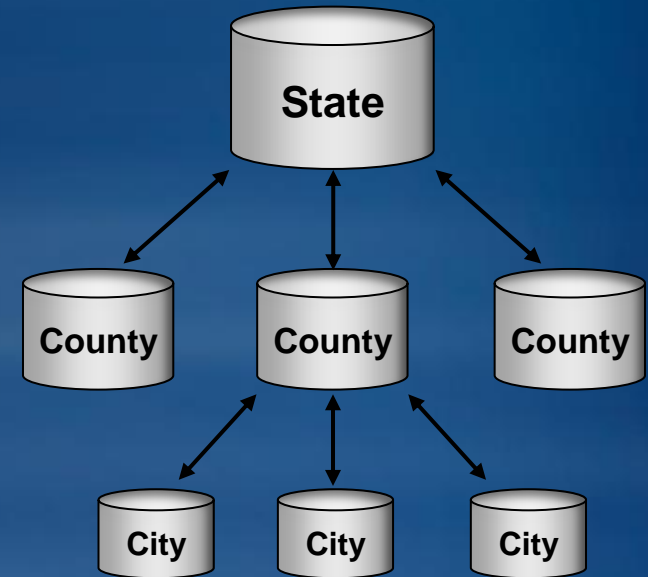
Fundamentals of Geodatabase Replication

- **User defines data to replicate from source geodatabase**
- **Replica describes data and how to synchronize changes**
  - **Parent replica**
  - **Child replica**
- **Data edited in versioned environment**
- **Synchronize changes**
  - **Send to related geodatabase**
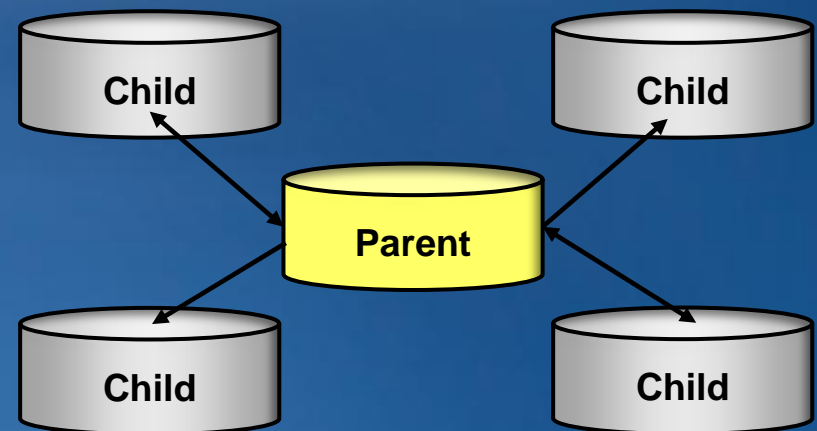
Types of replication

- **Single generation**
  - –One check out / check in operation

**Check out/
check in**

Parent — ArcSDE ← *Once Only* — Child — Any geodatabase

- **Multigeneration**
  - –Changes synchronized multiple times

One Way

Parent — ArcSDE → *Multiple Times* → Child — Any geodatabase

Two Way

Parent — ArcSDE ↔ *Multiple Times* ↔ Child — ArcSDE

Replica Type: Check-out/Check-in

- **Single round trip**
  - Check out once, check in once
  - Only one synchronization operation allowed
  - Conflicts detected on check in

- **Same as pre 9.2 disconnected editing**

- **Workflow scenarios**
  - Contractor delivering one set of updates at end of project
  - Mobile crew replicates small set of data for editing in the field

Parent (SDE) geodatabase

Check in

Check out
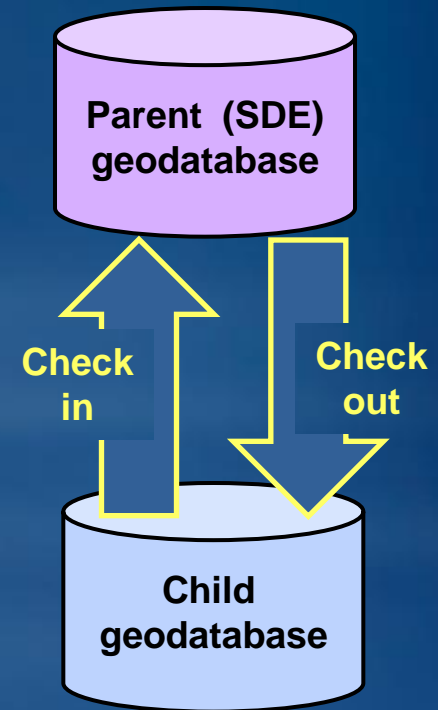
Child geodatabase

# Replica Type: One-way

- **Changes sent from parent to child**
- **Child geodatabase can be any type of geodatabase**
- **No conflict detection**
- **Workflow scenarios**
  - **Read-only publication database**
  - **requires updates from parent**

Parent (SDE) geodatabase

Sync changes **Create replica**
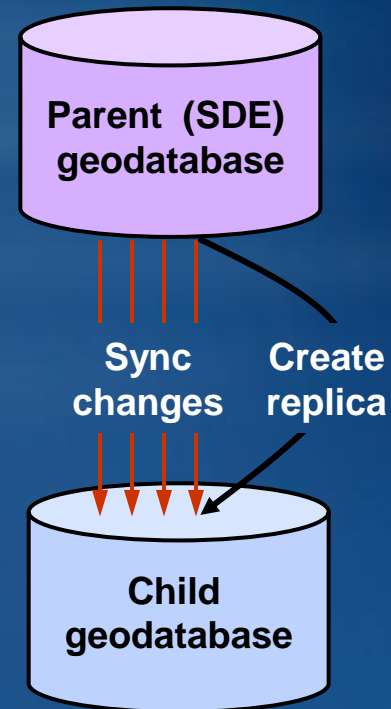
Child geodatabase

Replica Type: Two-way

- **Changes sent in both directions**
  - Parent to child; child to parent
  - Conflict detection used

- **Child geodatabase must be multiuser**

- **Workflow scenarios**
  - Data edited in different offices
    - Functionally or geographically separated data
  - Inconsistent or slow network speeds
    - Subsets of data in different location; send changes back and forth

Parent (SDE) geodatabase

Sync changes   **Create replica**
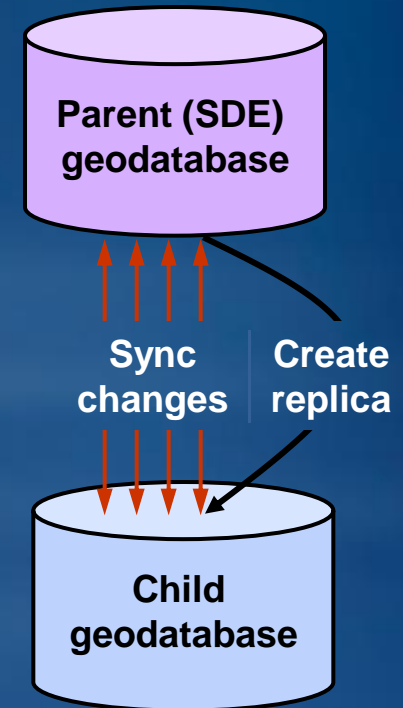
Child geodatabase

Replica Tree

- **Allows organizations to distribute data across several geodatabases in a hierarchical structure**
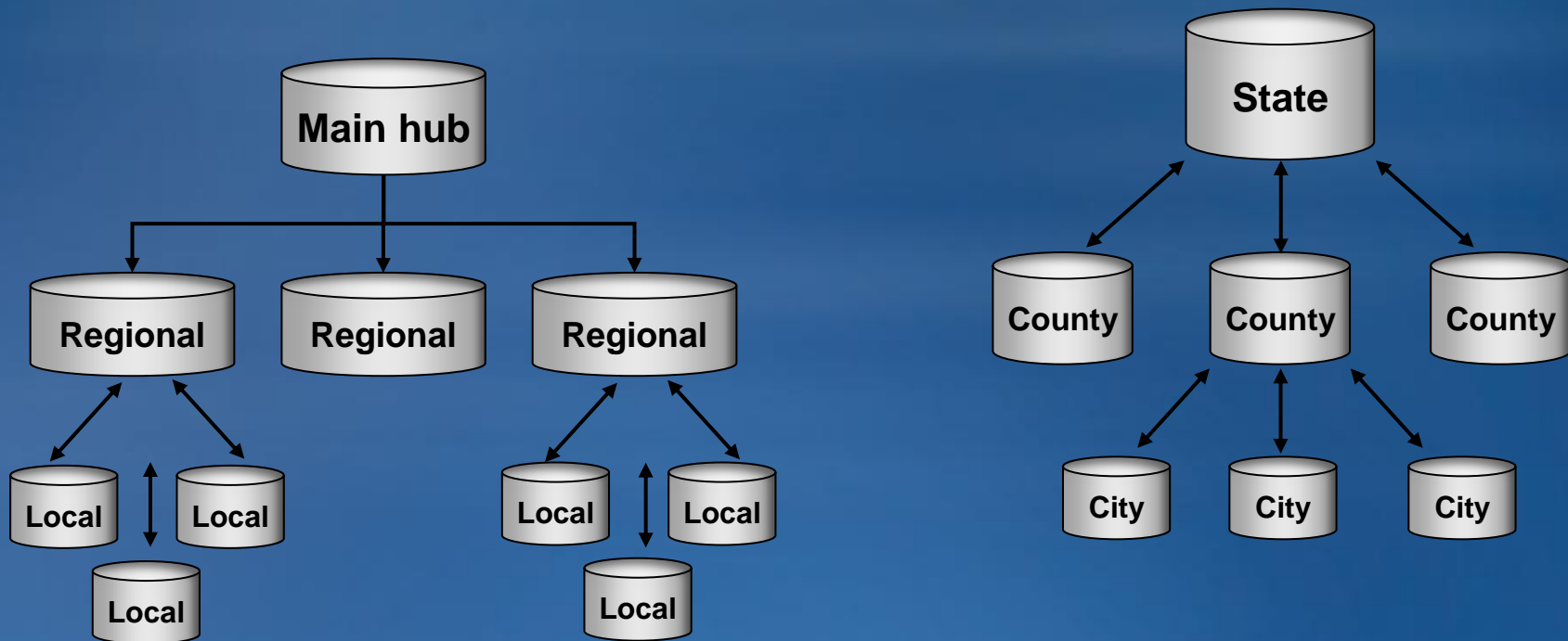
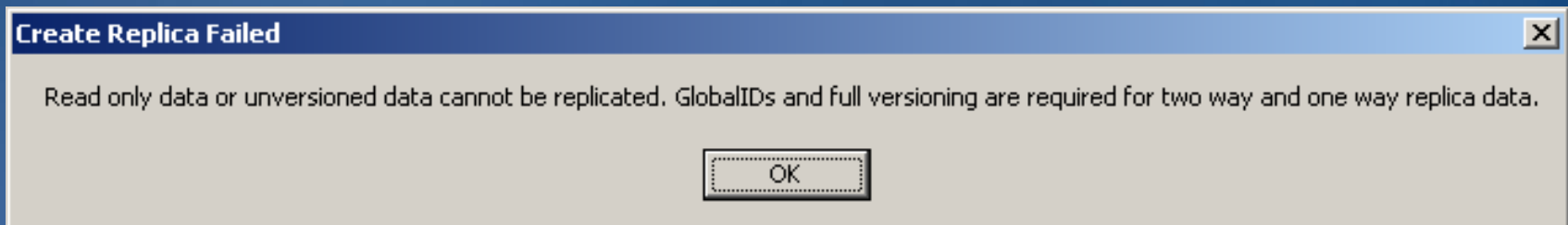# Maintaining Object Identity

- **Replicated object has both local and global identity**
  - **ObjectID** - Local identity is unique within a database
  - **GlobalID** - Global identity is unique across databases
- **GlobalID columns**
  - Based on GUIDs technology
    {9DFACA0A-982F-4175-80E7-B553378D9E6D}
  - GlobalIDs are system maintained

Creating a Replica

Replica Creation

- **All replicas**
  - Read/Write on data
  - Read on version
  - Stored in multiuser gdb
  - Data registered as versioned
  - Cannot use move to base

- **One-way and two-way**
  - Global Unique IDs required
  - Data uses high precision

Create Replica Failed

Read only data or unversioned data cannot be replicated. GlobalIDs and full versioning are required for two way and one way replica data.

OK

- **Filters**
  - Create filters before running Check-out wizard
  - Customized for each object
- **Check-out related objects to preserve relationships**

Demonstration

Create a Replica

Replica Synchronization

Synchronization Occurs through Messaging

- **Replica pairs synchronize through messages**
  - Data change message—includes actual data changes
    - All changes since last acknowledgement
  - Acknowledgement message
    - Data receiver acknowledges receipt of changes
  - Switch roles message
    - Data receiver now can send changes to replica relative (2 way)
  - If a data changes message is lost in a disconnected system, the next message will contain changes from the lost message and any new changes
- **Connected synchronization (connected on same network)**
  - ArcGIS manages messages
- **Disconnected synchronization (not on same network)**
 - Manual messaging required

Connected Synchronization

- Can be performed when all replicas are accessible on the network (LAN or WAN)
- Always connected or intermittently connected
- System manages message exchange and always acknowledges
- Can send changes in one direction or both directions in one operation
- Use the synchronize wizard in ArcCatalog/ArcMap

Synchronize

Disconnected Synchronization

- **Data sender exports data change message**
  - Delta geodatabase or XML file created with data changes
  - Transferred via CD or email
  - Child imports data change message
- **Data receiver exports acknowledgement message**
  - XML or delta geodatabase
  - Parent imports acknowledgement message
- **Switch role message**
  - Allows replica pair to change sender/receiver roles

Data change message →

Data sender
Replica version

← Acknowledgement message

Switch role message →

Data receiver
DEFAULT
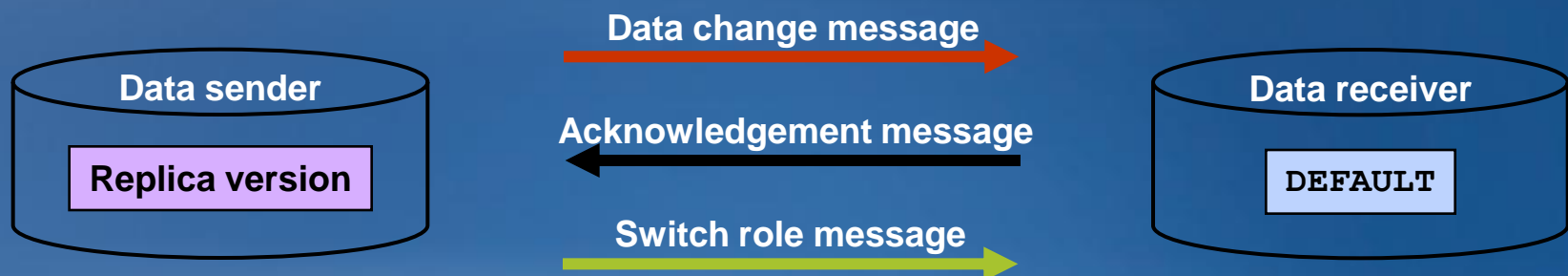
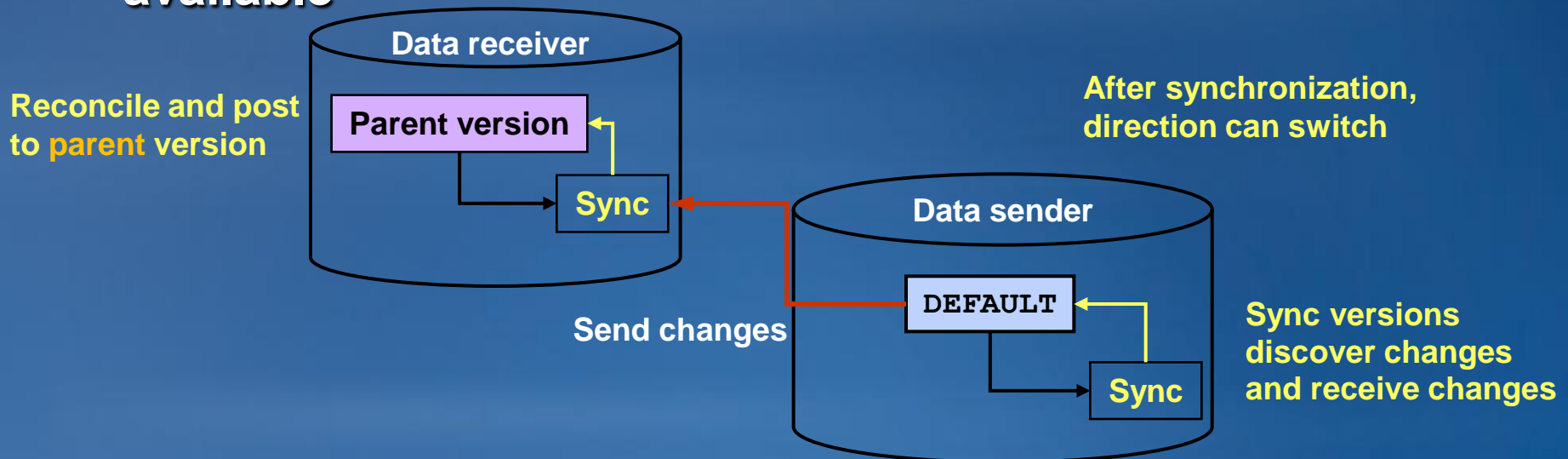# Workflow Scenario:
Two Way Replication

Create replica: Two-way

- **Connect to version to replicate— parent replica version**
- **Child replica version always `DEFAULT`**
- **Synchronization versions**
  - Hidden versions—created on both parent and child replicas
  - Used to identify changes to send
  - Used to receive changes for reconcile and post to replica version

Parent replica

Connect to parent (replica) version

Replica version

Sync

Replicate

Child replica

DEFAULT

Sync

Replicate to DEFAULT
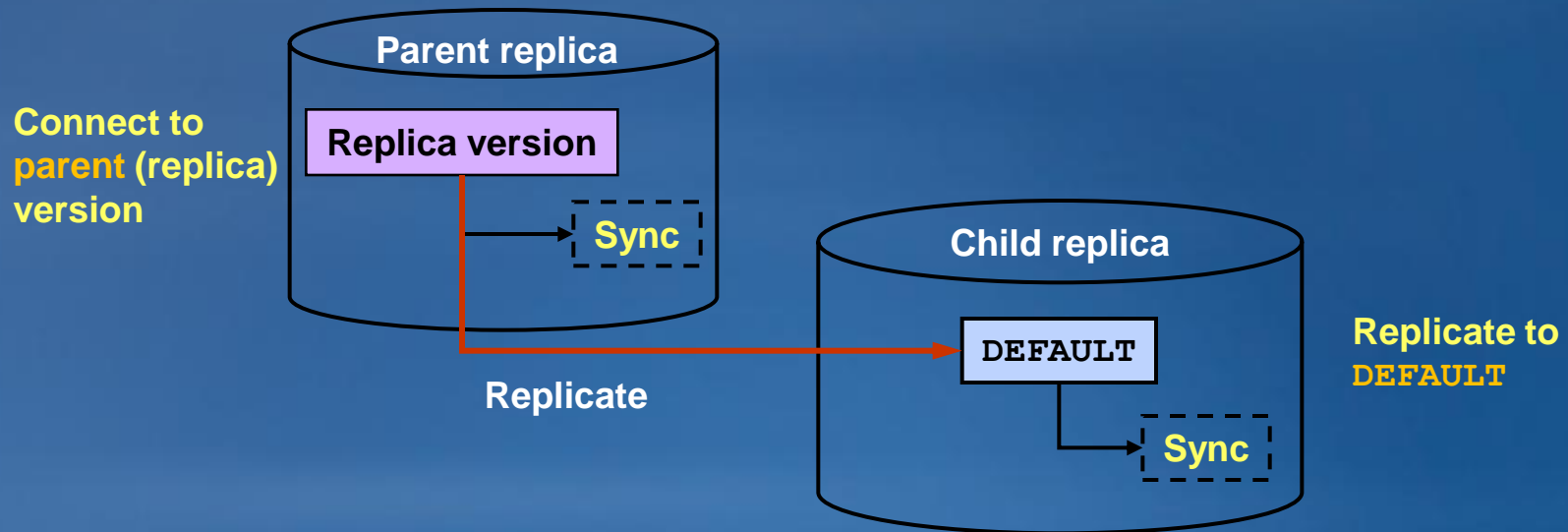
Synchronization: Two-way

- **Either member of replica pair can send changes**
  - Data sender sends changes; data receiver receives changes
- **Synchronization versions send and receive changes**
  - Detect differences to send to replica version
  - Receive differences to reconcile and post with replica version
  - Conflicts detected—automatic conflict management available

Data receiver

Reconcile and post to parent version

Parent version

Sync

After synchronization, direction can switch

Data sender

DEFAULT

Send changes

Sync

Sync versions discover changes and receive changes

Replica Reconcile Policies

- **Three policies:**
  - Favor the database – if there is a conflict, the value currently in the replica database is applied
  - Favor the imported changes – if there is a conflict, the value being imported is applied
  - Manual
    - When a conflict occurs, the replica is marked as in conflict. While in conflict it can receive changes but not send changes
    - To take it out of a conflict state, you need to later manually reconcile and post the synchronization version with the replica version
- **When synchronizing in both directions at once, manual can not be used as a synchronization policy**
- **Default is for parent replica's representation to be used**

Demo: Replica Synchronization

Working with Schema Changes
(avoid if possible)

- **Fault tolerant**
  - In most cases synchronization will still execute successfully even if each replica makes schema changes
  - Example: If a field has been dropped, synchronization skips that field
  - Could result in data loss, invalid values
- **Tools to apply schema changes across replicas**
  - Subset of schema changes can be applied
  - Recommend applying schema changes across all replicas before synchronizing
  - Schema changes can be applied in both connected and disconnected environments

Managing Replicas

- **Replica Manager Utility**
  - Rename
  - Unregister
  - View **log**
  - Refresh
  - Validate schema
  - Review properties
  - Remove datasets from a replica

- Available in ArcMap and ArcCatalog
- Accessed from the Distributed Geodatabase Toolbar

Replica Log

- **Available for one-way and two-way replicas**
- **Maintains a record of data messages sent and received by the replica**
- **Log contains:**
  - **Log Date – date and time of messages sent or received**
  - **Event – whether the message was sent or received by this replica**
  - **Result – if there is a conflict, view error report**
    - *Error report describes why there is a conflict and details the steps involved to remove the replica from its conflict state*
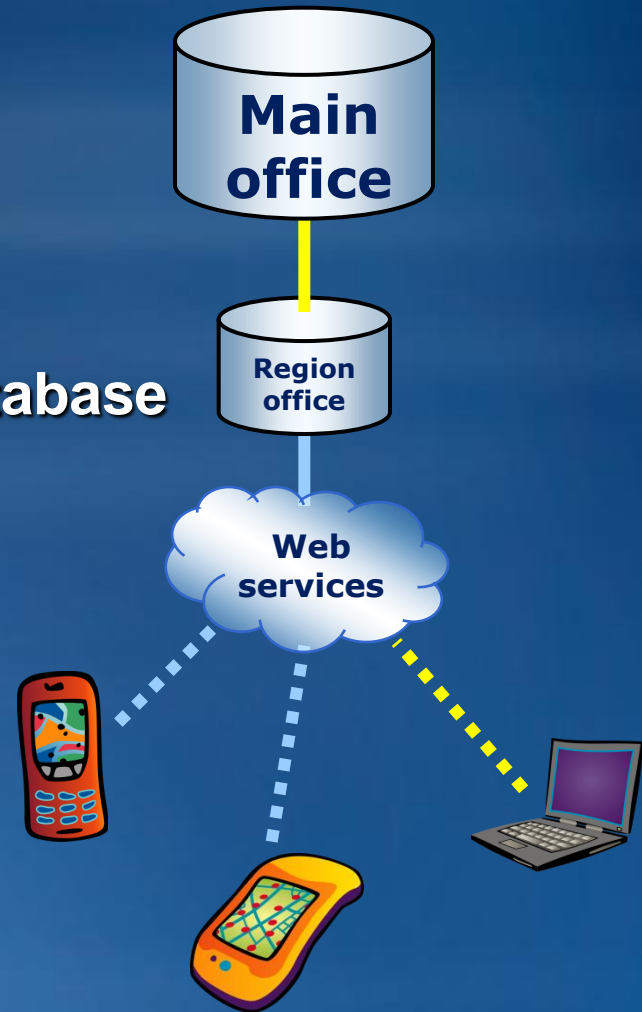
Geodata Services

Data distribution in Enterprise systems

- **Geodata services can be used in conjunction with other data distribution techniques**
- **Scenario**
  - Use geodatabase replication to synchronize changes between offices
  - Use Mobile services for field workers with lightweight mobile devices
  - Use geodata services for field workers who need ArcGIS Desktop or ArcGIS Engine in the field

Main office

Region office
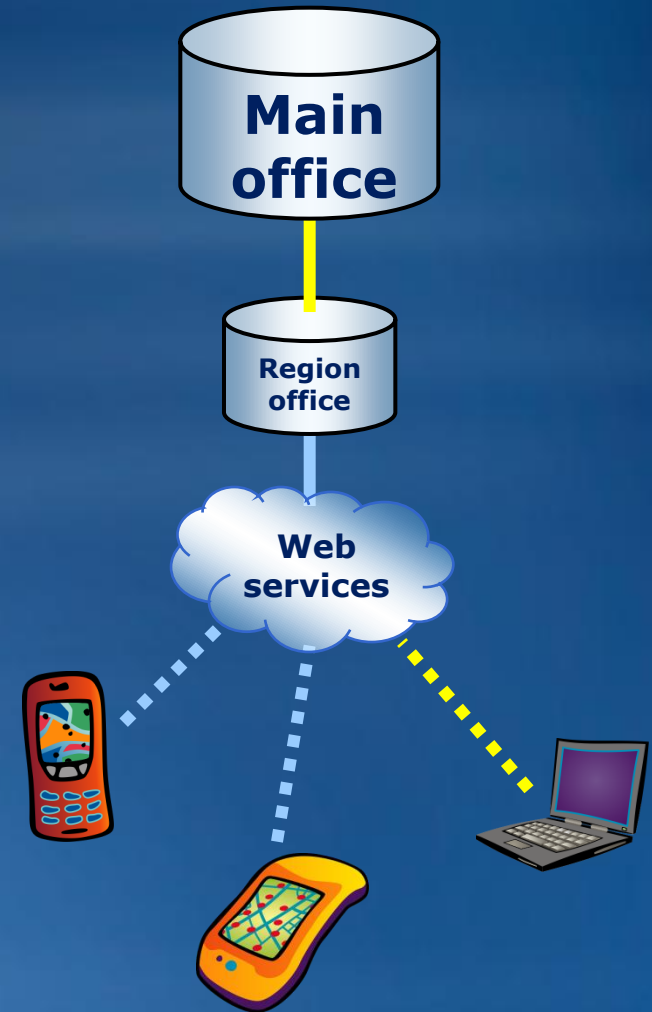
Web services
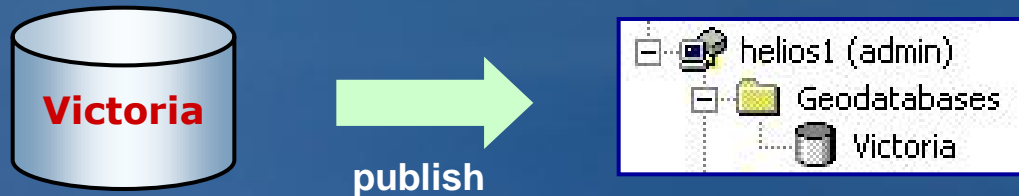
What are Geodata Services?

- **Provides remote access of geodatabases over over the web using ArcGIS Server**
    - Supports all types of geodatabases

- **Enables the following functionality:**
    - Create local copies of remote data
    - Execute geodatabase queries
    - Synchronize edits with another geodatabase (a.k.a. Geodatabase replication)

Main office

Region office

**Web services**

2 ways to Publish Geodata Services

1. **Directly from a geodatabase**
   - Database connection file (.sde), personal or file geodatabase
   - Note: SQLExpress databases cannot be published directly – publish via a map service

Victoria → **publish** → helios1 (admin) / Geodatabases / Victoria

2. **From a map document with geodata access enabled**
   - Creates 2 services: map service & geodata service
   - E.g., Use Create replica wizard in ArcMap

Ottawa.mxd → **publish** → helios1 (admin) / Geodatabases / Victoria / Maps / Ottawa / Ottawa
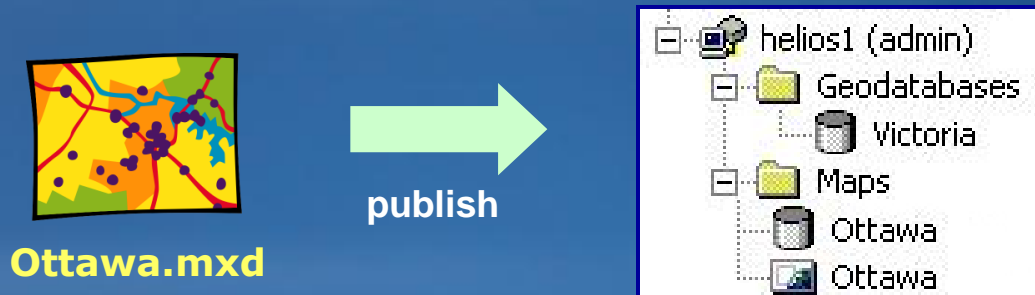
Publishing a Geodatabase

- **Choose operations allowed**
  - **Extraction – data can be copied from the geodatabase**
  - **Query – queries can be executed against the geodatabase (requires custom clients)**
  - **Replication (ArcSDE geodatabases only) – edits can be synchronized with another geodatabase**
- **Creates a single geodata service**
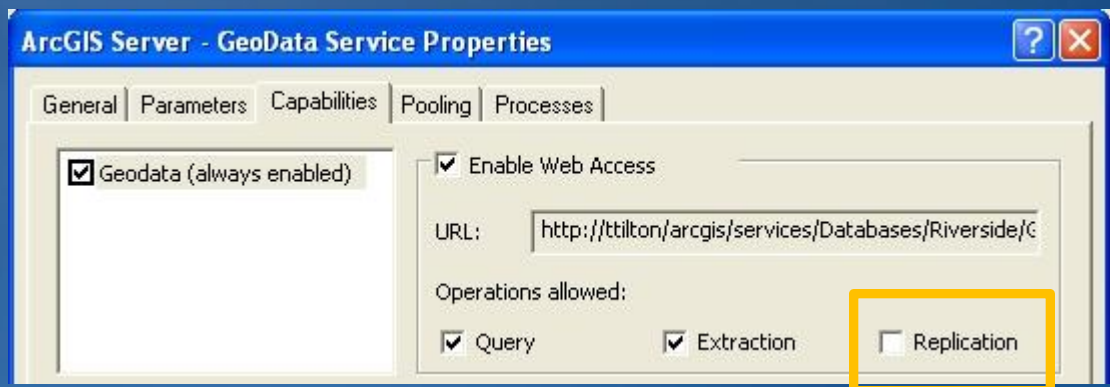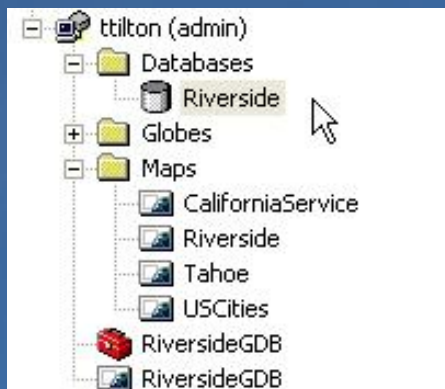
Publishing a Geodata Service from a Map

- **Make sure map contains geodatabase data**
  - **From only one geodatabase**
- **Enable the Geodata Access capability**
- **Creates two services with the same name**
  - **Geodata service**
  - **Map service**
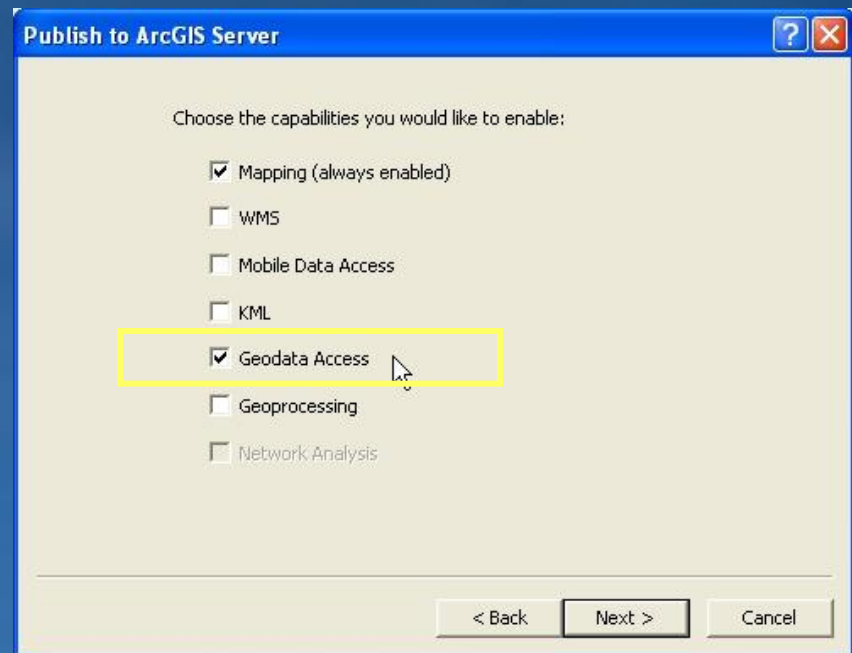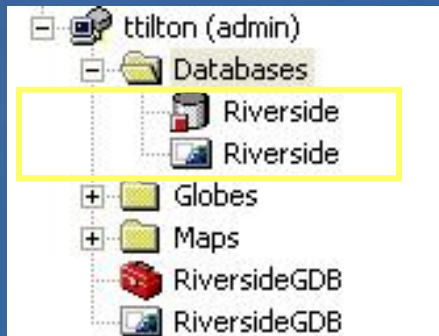
Thank you!!

Questions??

Heather Paskevic
hpaskevic@esri.com